

CLAIMS

- 1 1. A method for converting a file access data structure from a first endianness to a
2 second endianness, the method comprising the steps of:
3 identifying, from a descriptor look up table, a series of actions to perform on ele-
4 ments of the file access data structure; and
5 performing the identified series of actions on the elements of the file access data
6 structure.
- 1 2. A method of converting elements of a file access data structure from a first endi-
2 anness to a second endianness, the method comprising the steps of:
3 determining if the file access data structure is a critical path data structure;
4 converting, in response to the file access data structure being a critical path data
5 structure, the elements from the first endianness to the second endianness using a set of
6 specific code functions;
7 converting, in response to the file access data structure not being a critical path
8 data structure, a header of the file access data structure from the first endianness to the
9 second endianness using a second set of specific code functions; and
10 calling a byte swapping engine to convert selected elements of the file access data
11 structure from the first byte order to the second byte order.
- 1 3. The method of claim 2 wherein the file access data structure further comprises a
2 direct access file access data structure.
- 1 4. A system for converting elements of a file access data structure from a first endi-
2 anness to a second endianness, the system comprising:
3 an input buffer, the input buffer storing the file access data structure to be con-
4 verted;
5 a byte swapping engine, the byte swapping engine operative interconnected with a
6 descriptor table; and

7 an output buffer, the byte swapping engine placing the file access data structure in
8 the output buffer after conversion.

1 5. The system of claim 4 wherein the descriptor table further comprises a set of en-
2 tries describing various file access data structures, each entry further comprising a size
3 field and an operation field.

1 6. The system of claim 4 wherein the file access data structure further comprises a
2 direct access file access data structure.

1 7. A method for converting a data structure from a first byte order to a second byte
2 order, the method comprising the steps of:
3 reading an element entry from a descriptor table;
4 performing an action on an element of the data structure, the action being defined
5 in the element entry read from the descriptor table; and
6 placing the element in an output buffer.

1 8. The method of claim 7 wherein the step of performing an action on an element
2 further comprises the step of copying the element from an input buffer to the output
3 buffer.

1 9. The method of claim 7 wherein the step of performing an action on an element
2 further comprises the step of byte swapping the element.

1 10. The method of claim 7 wherein the element entry of the descriptor table further
2 comprises a field describing a size of the element and a field describing an action to be
3 performed.

1 11. A file server for use in a network storage environment, the file server comprising:
2 a byte swapping engine, the byte swapping engine performing a defined operation
3 on each of a plurality of elements of a file access data structure.

1 12. The file server of claim 11 wherein the file server further comprises a descriptor
2 look up table, the descriptor look up table having a plurality of entries, each of the plural-
3 ity of entries associated with a specific file access data structure.

1 13. The file server of claim 12 wherein each of the plurality of entries further com-
2 prises a plurality of elements, each of the elements having a size field and an operation
3 field.

1 14. The file server of claim 13 wherein the defined operation is defined by the opera-
2 tion field of the entry associated with the file access data structure.

1 15. A computer-readable medium, including program instructions executing on a
2 computer, for converting elements of a file access data structure from a first endianness
3 to a second endianness, the method comprising the steps of:
4 determining if the file access data structure is a critical path data structure;
5 converting, in response to the file access data structure being a critical path data
6 structure, the elements from the first endianness to the second endianness using a set of
7 specific code functions;
8 converting, in response to the file access data structure not being a critical path
9 data structure, a header of the file access data structure from the first endianness to the
10 second endianness using a second set of specific code functions; and
11 calling a byte swapping engine to convert selected elements of the file access data
12 structure from the first byte order to the second byte order.

1 16. A method for converting elements of a file access data structure from a first endi-
2 anness to a second endianness, the method comprising the steps of:
3 determining a type of the file access data structure;
4 processing, in response to the file access data structure of being of a first type, the
5 file access data structure along a first processing path;
6 processing, in response to the file access data structure being of a second type, the
7 file access data structure along a second processing path.

- 1 17. The method of claim 16 wherein the first type further comprises a critical path
2 data structure.
- 1 18. The method of claim 16 wherein the first processing path further comprises a set
2 of specifically coded functions.
- 1 19. The method of claim 16 wherein the second processing path further comprises a
2 byte swapping engine.

1006676-024903
000000-000000